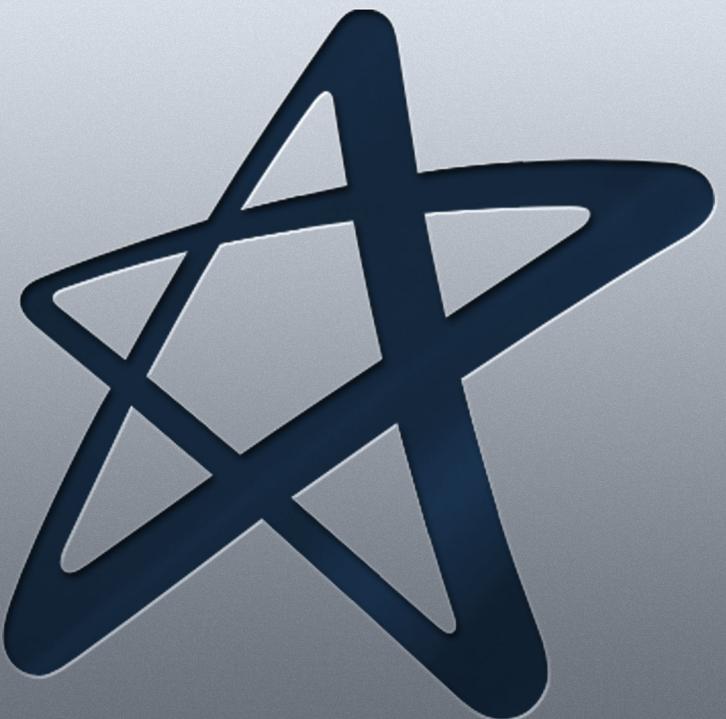


Banco de Dados



Educação a Distância
Cruzeiro do Sul Educacional
Campus Virtual

Material Teórico



Cláusula GROUP BY e HAVING

Responsável pelo Conteúdo:

Prof. Ms. Alexander Gobbato Albuquerque

Revisão Textual:

Profa. Esp. Vera Lídia de Sá Cicarone



- Conceito
- Criando Grupo de Dados



Objetivo de APRENDIZADO

- Funções de Grupo
- Revisão sobre funções de Grupo
- Cláusula GROUP BY
- Cláusula HAVING

Lembramos a você a importância de realizar todas as atividades propostas dentro do prazo estabelecido para cada unidade. Dessa forma, evitará que o conteúdo se acumule e que você tenha problemas ao final do semestre.

Uma última recomendação: caso tenha problemas para acessar algum item da disciplina ou dúvidas com relação ao conteúdo, não deixe de entrar em contato com seu professor tutor através do botão mensagens.

Contextualização

A instrução SELECT possui duas cláusulas poderosas, pouco compreendidas e usadas: GROUP BY e HAVING.

A cláusula GROUP BY organiza dados em grupos, produzindo sumários. A cláusula HAVING estabelece condições para listar esses grupos. Dizemos que a cláusula HAVING está para a cláusula GROUP BY assim como a cláusula WHERE está para o comando SELECT.

Conceito



A funcionalidade das funções de grupo é a de resumir informações, permitindo que sejam obtidas através de grupos de linhas com o uso de grupos ou agrupamento de funções .

EMPLOYEES

DEPARTMENT_ID	SALARY
90	24000
90	17000
90	17000
60	9000
60	6000
60	4200
50	5800
50	3500
50	3100
50	2600
50	2500
80	10500
80	11000
80	8600
	7000
10	4400

Salário máximo da tabela EMPLOYEES

MAX(SALARY)
24000

Revisão Função de Grupo

As funções de grupos estão listadas abaixo:

Função	Descrição
SUM	Retorna a soma de N.
AVG	Retorna a média aritmética de N.
COUNT	Retorna o número de linhas da consulta.
MAX	Retorna o valor máximo de N.
MIN	Retorna o valor mínimo de N.

Todas as funções acima operam sobre um número de linhas (por exemplo, uma tabela inteira) e são, portanto, funções de GRUPO.

DISTINCT faz uma função de grupo considerar valores não duplicados,

ALL considera todos os valores e sua declaração não é necessária.

Os tipos de dados dos argumentos devem ser alfanuméricos, numéricos ou data em que a expressão é listada.

Todas as funções de grupo, exceto COUNT(*), ignoram os valores nulos.

AVG

Retorna a média aritmética de um grupo de registros.

Para calcular a média salarial dos empregados, faça:

```
SELECT AVG(Salary)  
FROM Employees;
```

AVG(SALARY)
6461.68224299065420560747663551401869159

Note que as linhas da tabela Employees são trilhadas num único grupo (uma única linha).

Min – Max – Sum

Retornam o menor valor de um grupo de registros.

Uma função de grupo pode ser usada para subconjunto de linhas de uma tabela usando a cláusula WHERE.

Para encontrar o menor salário, o maior salário e a soma dos salários dos funcionários do departamento 30, faça:

```
SELECT MIN(Salary), MAX(Salary), SUM(Salary)  
FROM Employees  
WHERE Department_ID = 30;
```

MIN(SALARY)	MAX(SALARY)	SUM(SALARY)
2500	11000	24900

Count

Retorna a quantidade de registros de um grupo de registros.

Para encontrar o número de empregados do departamento 30, faça:

```
SELECT Count(*)  
FROM Employees  
WHERE Department_ID = 30;
```

COUNT(*)
6

Criando Grupo de Dados



Cláusula GROUP BY

EMPLOYEES

DEPARTMENT_ID	SALARY
10	4400
20	13000
20	6000
50	5800
50	3500
50	3100
50	2500
50	2600
60	9000
60	6000
60	4200
80	10500
80	8600
80	11000
90	24000
90	17000

Média de salário da tabela EMPLOYEES para cada departamento

DEPARTMENT_ID	AVG(SALARY)
10	4400
20	9500
50	3500
60	6400
80	10033.3333
90	19333.3333

20 rows selected.

OBS: todos os campos no SELECT que não possuírem função de grupo DEVEM APARECER NA CLÁUSULA GROUP BY.

```
SELECT department_id, AVG(salary)
FROM employees
GROUP BY department_id;
```

Porém os campos existentes na cláusula GROUP BY não precisam aparecer no SELECT.

```
SELECT AVG(salary)
FROM employees
GROUP BY department_id;
```

Para calcular a média salarial de cada grupo de cargo, faça:

```
SELECT      department_id, AVG(salary)
FROM        employees
GROUP BY    department_id;
ORDER BY    department_id;
```

Agrupando mais que uma coluna

EMPLOYEES

DEPARTMENT_ID	JOB_ID	SALARY
90	AD_PRES	24000
90	AD_VP	17000
90	AD_VP	17000
60	IT_PROG	9000
60	IT_PROG	6000
60	IT_PROG	4200
50	ST_MAN	5800
50	ST_CLERK	3500
50	ST_CLERK	3100
50	ST_CLERK	2600
50	ST_CLERK	2500
80	SA_MAN	10500
80	SA_REP	11000
80	SA_REP	8600
20	MK_REP	6000
110	AC_MGR	12000
110	AC_ACCOUNT	8300

Adiciona o
salário na
tabela EMPLOYEES
para cada cargo
agrupado por
departamento

DEPARTMENT_ID	JOB_ID
10	AD_ASST
20	MK_MAN
20	MK_REP
50	ST_CLERK
50	ST_MAN
60	IT_PROG
80	SA_MAN
80	SA_REP
90	AD_PRES
90	AD_VP
110	AC_ACCOUNT
110	AC_MGR
	SA_REP

20 rows selected.

```
SELECT      department_id, dept_id, job_id  
SUM(salary)  
FROM        employees  
GROUP BY    department_id; job_id;
```

Restrições

- não se usa a cláusula WHERE para restringir grupos;
 - para restringir grupos, usa-se a cláusula HAVING;
 - não se pode usar funções de grupo na cláusula WHERE.

```
SELECT      department_id, AVG(salary)
FROM        employees
WHERE       AVG(salary) > 8000
GROUP BY    department_id;
```

```
WHERE      AVG(salary) > 8000
          *
ERROR at line 3:
ORA-00934: group function is not allowed here
```

Excluindo linhas quando estiver usando o GROUP BY

Linhos devem ser excluídas com a cláusula WHERE antes da divisão por grupos.

Para mostrar a média salarial para cada cargo, excluindo os departamentos 10, 20 e 30, faça:

```
SELECT      department_id, AVG(salary)
FROM        employees
WHERE       department_id not in (10,20,30)
GROUP BY    department_id;
ORDER BY    department_id;
```

Para mostrar a média salarial e a soma dos salários por cada departamento e cargo, faça:

```
SELECT      department_id, job_id, AVG(salary), SUM(salary)
FROM        employees
GROUP BY    department_id, job_id
ORDER BY    department_id;
```

DEPARTMENT_ID	JOB_ID	AVG(SALARY)	SUM(SALARY)
10	AD_ASST	4400	4400
20	MK_MAN	13000	13000
20	MK_REP	6000	6000
30	PU_CLERK	2780	13900
30	PU_MAN	11000	11000
40	HR REP	6500	6500
50	SH_CLERK	3215	64300
50	ST_CLERK	2785	55700
50	ST_MAN	7280	36400
60	IT_PROG	5760	28800
More than 10 rows available. Increase rows selector to view more rows.			

Restringindo o Resultado por Grupo

EMPLOYEES

DEPARTMENT_ID	SALARY
90	24000
90	17000
90	17000
60	9000
60	6000
60	4200
50	5800
50	3500
50	3100
50	2600
50	2500
80	10500
80	11000
80	8600
20	6000
110	12000
110	8300

o salário máximo
por departamento
quando for maior
que \$ 10.000

DEPARTMENT_ID
20
80
90
110

```
SELECT      department_id, MAX(salary)
FROM        employees
GROUP BY    department_id
HAVING     MAX(salary)>10000 ;
```

20 rows selected.

A cláusula HAVING

Use a cláusula HAVING se você quiser especificar qual grupo será mostrado.

Para mostrar a média salarial de todos os departamentos que tenham a média maior que 6000, faça:

Relembrando...

- A cláusula WHERE é condição para o SELECT.
- A cláusula HAVING é condição para um GROUP BY.

```
SELECT      department_id, AVG(salary)
FROM        employees
GROUP BY    department_id
HAVING      AVG(salary) > 6000
```

DEPARTMENT_ID	AVG(SALARY)
20	9500
40	6500
70	10000
80	8955.88235294117647058823529411764705882
90	19333.333333333333333333333333333333333333333
100	8600
110	10150
-	7000

Para mostrar a quantidade de empregados que tenham o salário acima de 10000 por departamento, faça:

```
SELECT      department_id, count(*)
FROM        employees
GROUP BY    department_id
HAVING      MAX(salary) > 10000
ORDER BY    department_id
```

DEPARTMENT_ID	COUNT(*)
20	2
30	6
80	34
90	3
100	6
110	2

Nota:



A cláusula HAVING deve ser colocada depois da cláusula GROUP BY e é utilizada para estabelecer condições dentro das funções de grupo.
A cláusula WHERE não pode ser usada para restringir itens de grupo.

A seguinte declaração da cláusula WHERE é errada.

```
SELECT      department_id, count(*)
FROM        employees
WHERE       MAX(salary) > 10000
GROUP BY    department_id
ORDER BY    department_id;
```



ORA-00934: group function is not allowed here

Você pode unicamente usar WHERE para restringir linhas individuais. Para restringir colunas de grupos, usa-se a cláusula HAVING.

Você pode limitar sua seleção incluindo somente os departamentos 10, 20, 30 e 40, usando a cláusula WHERE, quando estiver agrupando por departamento.

```
SELECT      department_id, count(*)
FROM        employees
WHERE       department_id IN (10,20,30,40)
GROUP BY    department_id
HAVING     MAX(salary) > 10000
ORDER BY    department_id;
```

DEPARTMENT_ID	COUNT(*)
20	2
30	6

A ordem das cláusulas na declaração SELECT.

```
SELECT      coluna(s)
FROM        tabela(s)
WHERE       condição linha
GROUP BY    coluna(s)
HAVING     condição de grupo de linhas
ORDER BY    coluna(s);
```

Material Complementar



Acesse o documento [mat_comp.pdf](#), é exibida a forma de manipular o dicionário de dados.

Referências

COSTA, Rogério Luis de C. **SQL: guia prático.** 2. ed. Rio de Janeiro : Brasport, 2006.

FANDERUFF, Damaris. **Dominando o Oracle 9i: Modelagem e desenvolvimento.** São Paulo: Pearson Education do Brasil, 2003.

SILBERSCHATZ, A. **Sistema de bancos de dados.** São Paulo: Pearson Education do Brasil, 2004.

Anotações





Educação a Distância
Cruzeiro do Sul Educacional
Campus Virtual

www.cruzeirodosulvirtual.com.br
Campus Liberdade
Rua Galvão Bueno, 868
CEP 01506-000
São Paulo SP Brasil
Tel: (55 11) 3385-3000



Universidade
Cruzeiro do Sul



UNICID
Universidade
Cidade de S. Paulo



UNIFRAN
Universidade
de Franca



UDF
Centro
Universitário



Módulo
Centro
Universitário